

Rancang Bangun Alat Ukur Kenyamanan Ruang (Termal dan Visual) Berbasis Arduino Uno

Marhabsyah Zhaki¹, Yudi Chadirin^{1*} dan Satyanto Krido Saptomo¹

¹ Departemen Teknik Sipil dan Lingkungan, Fakultas Teknologi Pertanian, Institut Pertanian Bogor, Bogor, Indonesia 16680

* penulis koresponden: yudi@apps.ipb.ac.id

Abstrak: Kenyamanan ruang dapat mempengaruhi perilaku dan psikologis penggunanya. Kenyamanan ruang yang perlu untuk diperhatikan diantaranya adalah kenyamanan termal dan kenyamanan visual. Masyarakat terbiasa menggunakan alat ukur suhu berupa termometer dan alat ukur intensitas cahaya berupa lux meter untuk mengetahui kondisi ruang. Alat ukur tersebut menghasilkan besaran suhu dan intensitas cahaya yang terukur tanpa disertai justifikasi apakah memenuhi baku mutu atau tidak. Tanpa mengetahui pemenuhan baku mutu, tingkat kenyamanan termal dan visual tidak dapat ditentukan. Untuk itu diperlukan alat ukur yang dapat mengukur suhu dan intensitas cahaya sekaligus menentukan tingkat kenyamanannya berdasarkan kesesuaian dengan baku mutunya. Pada penelitian ini dilakukan perancangan alat ukur kenyamanan termal dan visual ruangan dengan menggunakan mikrokontroler arduino uno, yang dilengkapi dengan sensor DHT11 untuk mengukur suhu dan kelembaban relatif udara, sensor BH1750 untuk mengukur intensitas cahaya. Penentuan tingkat kenyamanan termal berdasarkan baku mutu yang tertera dalam SNI-03-6572-2001 dan baku mutu kenyamanan visual berdasarkan SNI-03-6575-2001. Hasil kalibrasi antara sensor dengan alat ukur standar yakni *environment meter* diperoleh hasil rata-rata persentase *error* yang berada dalam batasan wajar. Nilai rata-rata persentase *error* pembacaan sensor parameter suhu, kelembaban relatif dan intensitas pencahayaan tanpa *cover* berturut-turut bernilai 1.56%, 2.95%, dan 0.76%. Pada penggunaan tutup pada sensor menghasilkan nilai *error* berturut-turut 1.74%, 2.73%, 1.31%. Prototype ini membagi tingkat kenyamanan termal menjadi tiga yakni Sejuk Nyaman (SN), Nyaman Optimal (NO) dan Hangat Nyaman (HN). Jika suhu udara yang terbaca kurang dari 20.5 °C atau lebih tinggi dari 27.1 °C maka alat akan menampilkan sebagai kondisi Tidak Nyaman (TN). Kenyamanan visual terbagi menjadi 2 yakni Nyaman (NM) jika hasil pengukuran pada rentang 120 – 250 lux, di luar rentang nilai tersebut ditampilkan sebagai kondisi Tidak Nyaman (TN). Hasil tersebut menunjukkan alat *monitoring* yang dibangun dapat beroperasi dengan baik karena nilai pembacaan mendekati sebenarnya.

Kata kunci: arduino uno, kenyamanan ruangan, sensor BH1750, sensor DHT11

Diterima: 08 Desember 2022

Disetujui: 12 April 2023

Sitasi:

Zhaki, M.; Chadirin, Y.; Saptomo, S.K. Rancang Bangun Alat Ukur Kenyamanan Ruang (Termal dan Visual) Berbasis Arduino Uno. *J. Teknik Sipil dan Lingkungan*. 2023; 8 [1]: 57-66.,
<https://doi.org/10.29244/jsil.8.1.57-66>

1. Pendahuluan

Mikrokontroler pada masa ini menjadi salah satu teknologi yang mengalami perkembangan dan kemajuan yang begitu pesat. Sifat *opensource* yang dimiliki mikrokontroler memberikan kemudahan bagi manusia agar memaksimalkan pemanfaatan teknologi dalam meningkatkan produktivitas sehari-hari [1]. Penerapan teknologi mikrokontroler dapat diimplementasikan pada rumah tempat tinggal. Rumah layak dan nyaman jika dirancang dapat memenuhi kenyamanan penggunanya sehingga pengguna didalamnya merasa tenang, aman, dan nyaman akibat tingkat kenyamanan ruangan yang sesuai kebutuhan [2]. Kenyamanan ruangan

dapat memengaruhi perilaku dan psikologis penggunanya. Beberapa aspek kenyamanan ruangan yang perlu untuk diperhatikan adalah kenyamanan termal dan kenyamanan visual [3].

Kenyamanan termal timbul sebagai akibat dari termal atau panas serta perlakuan pada ventilasi atau bukaan dalam ruang sedangkan kenyamanan visual timbul sebagai akibat dari pencahayaan matahari maupun benda pemantul cahaya lainnya. Kedua parameter tersebut penting bagi kenyamanan ruangan sehingga dibutuhkan suatu alat ukur intensitas termal dan visual pada ruangan [4]. Alat ukur penentuan nilai termal (suhu dan kelembaban relatif) dan visual (intensitas pencahayaan) dapat menggunakan *environment meter* [5]. Alat tersebut memiliki kelebihan menggabungkan empat sensor pada satu alat yaitu *humidity sensor*, *temperatur meter*, *sound level*, dan *light meter*. Kekurangan alat ukur ini belum mampu menentukan tingkat kenyamanan ruangan secara otomatis berdasarkan pembacaan nilai pada alat. Akibat perkembangan mikrokontroler, penentuan tingkat kenyamanan ruangan secara *real time* dapat dilakukan dengan membutuhkan sebuah mikrokontroler dan sensor pendukung. Mikrokontroler yang dapat digunakan dan mudah diaplikasikan adalah mikrokontroler arduino uno.

Pengukuran nilai termal dapat digunakan sensor DHT11 dengan parameter suhu dan kelembaban yang terhubung dengan mikrokontroler arduino uno [6]. Nilai kenyamanan visual dapat menggunakan sensor BH1750 sebagai pendeteksi intensitas pencahayaan [7]. Berdasarkan penjelasan tersebut, studi ini dilakukan untuk merancang dan membuat sistem *monitoring* suhu, kelembaban relatif, dan intensitas pencahayaan ruangan serta menentukan kenyamanan termal dan kenyamanan visual pada ruangan menggunakan prinsip mikrokontroler arduino uno. Sehingga diharapkan mampu membandingkan kinerja sistem *monitoring* yang dibuat dengan alat ukur *environment meter*.

2. Metode

Penelitian ini dilaksanakan pada bulan April hingga Oktober 2022. Pada tahap awal dilakukan perakitan *prototype dan programming* arduino. Uji kalibrasi untuk membandingkan hasil pengukuran *prototype* dengan alat ukur *environmenter* dilakukan di dalam ruang terkontrol suhu dan kelembaban udara relatif menggunakan *growth chamber fitotoron* yang berlokasi di *Advanced Research Laboratory (ARLab)* IPB. Pengukuran kenyamanan dilakukan di ruang staf ARLab IPB dan kamar kos untuk mewakili ruangan kerja dan tempat tinggal. Pengukuran ini dilakukan sebagai unjuk kerja *prototype*.

2.1. Material

Bahan yang digunakan pada penelitian ini adalah rangkaian mikrokontroller yang meliputi komponen Arduino Uno ATmega328P, sensor suhu DHT11, sensor cahaya BH1750, modul I2C, ESP01, kabel, layer LCD dan *breadboard*.

2.2. Prosedur Penelitian

Secara umum tahapan penelitian meliputi desain *prototype arduino uno menggunakan* perangkat lunak *Proteus 8 Professional*, pengujian komponen, pembuatan *prototype* mikrokontroller Arduino uno, *Programming board arduino uno dan ESP01*, uji kinerja *prototype*, *penerapan aplikasi Blynk IoT*. Perancangan *prototype* pada *Proteus* mempermudah dalam pembuatan komponen yang akan digunakan sebelum realisasi komponen yang sebenarnya pada *prototype* alat. Tahapan awal yang dilakukan dalam mendesain *prototype* adalah dengan mengumpulkan komponen-komponen sesuai kebutuhan dan dirangkai menjadi satu kesatuan [8]. Pengujian komponen ini dilakukan secara terpisah pada tiap-tiap komponen dengan memasukkan *programming sederhana* yang diperoleh dari *library software Arduino IDE*. Adapun komponen-komponen *hardware prototype* yang diuji, yaitu sensor DHT11, sensor BH1750, mikrokontroler arduino uno, mikrokontroler ESP01, dan LCD 16 x 2. Tahapan pengujian kelima komponen alat tersebut diawali dengan menyiapkan *example coding* pada *software Arduino IDE* yang didapatkan setelah menambahkan *library*. Pembuatan *prototype* berupa perancangan serta evaluasi

terhadap desain yang telah dibuat [9]. Tahapan ini bertujuan untuk mengetahui apakah desain yang dirancang telah sesuai dengan tujuan pembuatan. Komponen-komponen alat yang sudah disiapkan dan sudah dilakukan pengujian, selanjutnya saling dihubungkan sesuai dengan desain). Tahapan *programming* dilakukan terhadap 2 jenis mikrokontroler yaitu arduino uno dan ESP01. Struktur perintah pada *software* ini, yaitu *header*, deklarasi variabel, *setup*, dan *loop*. Dalam menjalankan *software Arduino IDE* dibutuhkan *library*. Hasil dari *programming* ini dapat memberikan perintah kepada komponen lainnya yang terhubung dan diolah pada *board* mikrokontroler sehingga memunculkan tampilan pada LCD berupa nilai hasil pembacaan. Tahapan *programming* pada *board* ESP01 memiliki struktur yang sama dengan *programming board* arduino uno, yang menjadi perbedaan yaitu pada setiap strukturnya ditambahkan perintah baru agar terhubung dengan aplikasi *Blynk IoT*.

Tahap penentuan tingkat kenyamanan ruangan dilakukan dengan menambahkan perintah logika *if else* pada *programming*. Penentuan tingkat kenyamanan ruangan menggunakan standar mutu pada kelembaban udara yaitu SNI 6572-2001 tentang Tata Cara Perancangan Sistem Ventilasi dan Pengkondisian Udara pada Bangunan Gedung [10], serta intensitas cahaya menggunakan SNI 03-6575-2001 tentang Tata Cara Perancangan Sistem Pencahayaan Buatan pada Bangunan Gedung [11]. Penentuan menggunakan ASHRAE-55 diujikan pada alat *monitoring* untuk dibandingkan dengan penggunaan SNI. Uji kinerja *prototype* dilakukan dengan melakukan pengukuran di dalam ruang terkontrol (suhu, RH dan cahaya) dalam ruang fitotron secara bersamaan antara *prototype* dan alat ukur *environment meter*. Hasil pengukuran keduanya dibandingkan untuk menentukan persentase *error*. Penentuan persentase *error* digunakan rumus **Persamaan (1)**.

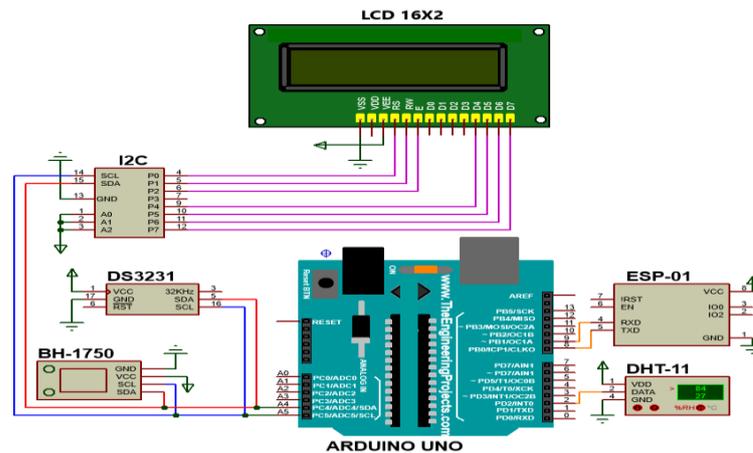
$$\%Error = \frac{|\text{Hasil baca } prototype - \text{Hasil baca alat lain}|}{\text{Hasil baca alat lain}} \times 100 \quad (1)$$

Uji kinerja *prototype* dilakukan dengan tambahan kabel (ekstensi) dan tanpa tambahan kabel. Untuk mengirimkan hasil pengukuran dari *prototype*, digunakan aplikasi *Blynk IoT*. Pembacaan melalui aplikasi *Blynk IoT* juga dapat dilihat melalui *website Blynk Cloud* dengan memasukkan *email* dan *password* yang sama pada aplikasi *Blynk IoT* di *smartphone*.

3. Hasil dan Pembahasan

3.1 Desain *prototype*

Sistem elektronika arduino uno dirancang menggunakan *software Proteus 8 Professional*. *Prototype* dibuat sesuai dengan kebutuhan sistem mikrokontroler yang dirancang [12]. *Software Proteus 8 Professional* menghasilkan simulasi yang menyerupai hasil sesungguhnya sehingga *software* ini digunakan dalam desain mikrokontroler sebelum membangun *hardware*. Namun pada penelitian ini, peran *software Proteus 8 Professional* hanya berguna untuk mendesain *prototype* karena tidak dimasukkan program perintah simulasi. Hal tersebut dikarenakan tidak ditemukannya *library* pada komponen-komponen yang dibutuhkan sehingga komponen alat dibuat secara manual. Komponen alat yang tidak tersedia pada *software* ini, yaitu ESP01, DHT11, BH1750, dan modul I2C. Penelitian ini dibagi menjadi dua proses perancangan alat yaitu meliputi perancangan perangkat lunak (*software*) dan proses implementasi perangkat keras (*hardware*). Hasil desain *hardware* direpresentasikan dalam model seperti tersaji pada **Gambar 1**.



Gambar 1. Desain *prototype* pada *software Proteus 8 Professional*

3.2 Implementasi desain *hardware*

Realisasi *prototype* merupakan implementasi desain yang telah dibuat, yaitu penerapan komponen *prototype* dan penyambungan antar pin komponen alat menggunakan kabel *jumper* [12]. Perancangan *hardware* secara garis besar adalah pembacaan sensor, pengolahan data, penampilan data, dan pengiriman data sensor. Rangkaian *hardware* diletakkan di dalam *box* elektronika hitam ukuran 18,5 cm x 11,5 cm x 6,5 cm. Sensor DHT11 dan BH1750 diletakkan di bagian atas *box*, LCD ditempatkan di depan, dan komponen-komponen lainnya berada di bagian dalam *box* elektronika. Mikrokontroler arduino uno sebagai mikroprosesor berfungsi untuk mengontrol sensor, pengolahan, penyimpanan dan pengiriman data. *Prototype* yang dibuat masih memiliki kekurangan yaitu tidak mampu menyimpan hasil pembacaan sensor secara otomatis. Hal ini disebabkan tidak adanya *memory* penyimpanan atau *SD Card* pada *prototype*. Hasil data pembacaan sensor dapat dilihat menggunakan aplikasi *Blynk IoT* dengan kewajiban berlangganan setiap bulannya.

Perangkat arduino menggunakan sistem *open source* secara *hardware* maupun *software* [13]. Dengan adanya metode *open source* ini maka pekerjaan bidang mikrokontroler dapat menjadi lebih mudah. Bahasa pemrograman mikrokontroler arduino uno menggunakan bahasa C. Bahasa C sudah dipermudah dengan tugas-tugas sederhana sehingga dapat diterapkan dengan cukup mudah [14].

Programming menggunakan *software* ini mampu melakukan pengambilan data, komputasi data, koreksi data. Struktur program *Arduino IDE* terbagi menjadi empat, yaitu *header*, deklarasi variabel, *setup*, dan *loop*. *Header* berfungsi dalam pemanggilan *library-library* yang dibutuhkan dalam penyusunan *coding*. Deklarasi variabel yaitu mendefinisikan tipe variabel, menyimpan, dan memindahkan karakter didalam program. *Setup* berisi kode perintah menentukan fungsi pada pin yang akan dibaca sekali oleh arduino uno. *Loop* adalah kebalikan dari *setup* yaitu dapat dibaca terus menerus oleh arduino uno setelah pembacaan pada *setup*. Tampilan pembacaan tingkat kenyamanan ruangan diatur pada *programming* bagian *void loop* dengan menambahkan logika *if else* diikuti dengan nilai standar kenyamanan termal dan visual berdasarkan SNI. Logika ini akan menjalankan fungsi menampilkan status tingkat kenyamanan ruangan pada LCD sesuai dengan ketentuan batasan nilai pembacaan oleh sensor. Logika diletakkan pada bagian *void loop* difungsikan mampu membaca nilai sensor secara terus-menerus untuk ditampilkan status kenyamanan ruangan di tampilan LCD. Tampilan tingkat kenyamanan pada LCD berupa keterangan mengenai kondisi kenyamanan termal yaitu suhu, kelembaban relatif dan kenyamanan visual.

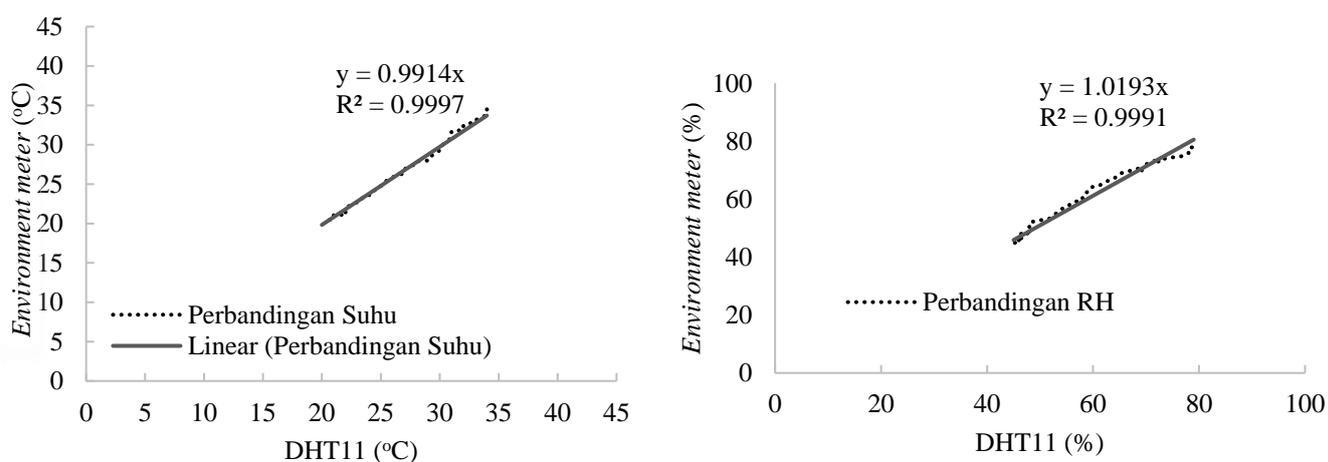
Programming pada board ESP01 pada dasarnya berisi perintah untuk ditampilkan hasil pembacaan pada aplikasi *Blynk IoT*. Perintah yang ditambahkan pada ESP01 diperoleh dari *library Blynk* yang didapatkan pada menu *manage library* di *software Arduino IDE*. Konsep untuk mendapatkan

pembacaan hasil yang sama dengan tampilan pada LCD setiap waktunya didasarkan dari perintah *string*. Perintah *string* berguna mengumpulkan hasil pembacaan sensor pada *serial monitor*, selanjutnya dikirimkan pada *board* ESP01 untuk menampilkannya pada *device* di aplikasi *Blynk* IoT. Pengiriman hasil pembacaan ini diperlukan jaringan internet yang terhubung dengan ESP01 sehingga penelitian ini termasuk menggunakan konsep *Internet of Things* (IoT). Proses penelitian ini berhasil menampilkan nilai pembacaan pada *prototype* dan aplikasi *Blynk* IoT

3.3 Hasil pembacaan kalibrasi

Tingkat keakuratan sensor DHT11 dan BH1750 dalam penelitian ini dibandingkan dengan alat ukur *environment meter*. Selanjutnya, untuk mendapatkan akurasi yang baik maka dilakukan kalibrasi hasil pengukuran *non-extension* dan dengan *extension*. Penelitian ini dibagi menjadi dua perlakuan dalam pengambilan data, yaitu sensor diberikan *extension* dan *non-extension*. Perpanjangan atau *extension* memiliki tujuan yaitu untuk menambah jangkauan pembacaan sensor. Perbandingan sensor DHT11 dan BH1750 dilakukan pengujian pertama dengan perlakuan *non-extension*. Tujuannya untuk didapatkan nilai pembacaan sensor tanpa adanya halangan yakni ketika menggunakan penutup. Pengujian dilakukan dengan memasukkan *prototype* dan *environment meter* ke dalam *thermal test chamber* yang telah diatur batasan termal di dalamnya.

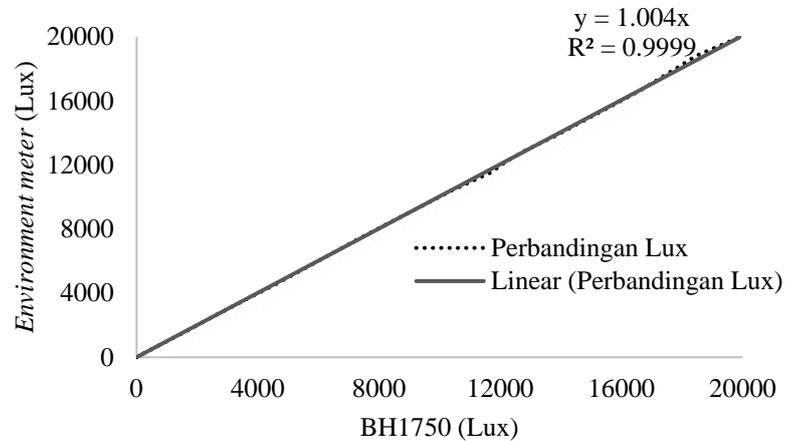
Pengambilan data yang dilakukan pada sensor DHT11 dengan parameter suhu dihasilkan nilai rata-rata selisih antara pembacaan oleh DHT11 dengan *environment meter* didapatkan nilai sebesar 0,37°C. Salah satu faktor yang menyebabkan ditemukannya selisih dari keduanya karena nilai pembacaan pada DHT11 tidak berbentuk desimal. Persamaan linearitas diperoleh yakni $y = 0,9914x$, artinya bernilai baik karena nilai a mendekati 1. Performa DHT11 didapatkan hasil bahwa lamanya waktu pembacaan sensor dari suhu 34°C – 20°C adalah 42 menit. Hasil pembacaan DHT11 dengan parameter kelembaban relatif memiliki waktu performa yang sama ketika dilakukan pembacaan RH oleh sensor pada nilai 48% - 79%. **Gambar 2** menyajikan grafik perbandingan kelembaban relatif *non-extension* dibandingkan dengan hasil pengukuran menggunakan alat ukur RH standar (*environment meter*).



Gambar 2. Perbandingan (a) suhu dan (b) RH sensor DHT11 *non-extension*

Kurva kalibrasi pembacaan RH di atas diperoleh persamaan linearitas yakni $y = 1,0193x$. Hasil pengambilan data diperoleh nilai rata-rata selisih pembacaan RH dari DHT11 dan alat ukur sebesar 1,83%. Faktor yang menyebabkan adanya nilai selisih tersebut diyakini akibat pintu *chamber* yang

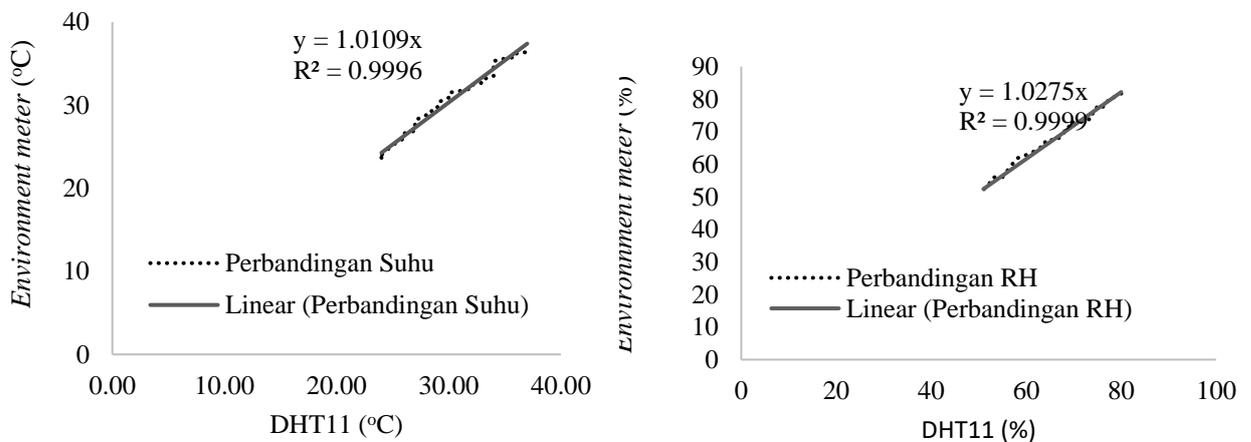
kadang kala dibuka tutup untuk menghidupkan kembali alat ukur yang tidak bisa konsisten menyala. Selanjutnya, kurva perbandingan BH1750 dapat dilihat pada **Gambar 3**.



Gambar 3. Perbandingan sensor BH1750 *non-extension*

Berdasarkan kurva pada **Gambar 3** diperoleh persamaan linearitas yakni $y = 1,004x$. Nilai rata-rata selisih yang didapatkan antara kedua alat sebesar 34.34 Lux. Nilai ini didapatkan cukup besar, faktor penyebabnya diyakini akibat kemampuan dari *environment meter* yang hanya dapat melakukan pembacaan maksimum di 20.000 Lux dan akibat pencahayaan yang diterima oleh sensor maupun alat ukur yang tidak sama. Pengaruh letak posisi sensor dan *environment meter* dinilai juga cukup berpengaruh terhadap nilai pembacaan karena adanya bayangan yang menutupi sensor dapat mengubah nilai pembacaan. Performa lamanya waktu pembacaan dari 0 Lux – 20.000 Lux oleh BH1750 yakni 1 menit 30 detik.

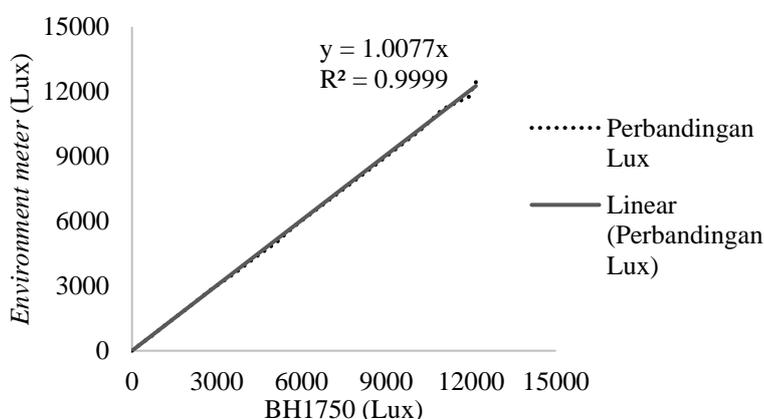
Tujuan digunakannya kabel ekstensi sensor yaitu untuk menambah jangkauan pembacaan sensor. Proses kalibrasi dilakukan juga menggunakan perbandingan *environment meter* dan diletakkan keduanya ke dalam *thermal test chamber*. Hasil pengukuran tersaji pada **Gambar 4**.



Gambar 4. Perbandingan suhu sensor DHT11 dengan *extension* (a) suhu (b) RH

Berdasarkan kurva perbandingan **Gambar 4** diperoleh persamaan linearitas $y = 1,0109x$. Pembacaan suhu oleh DHT11 pada rentang 24°C – 37°C memiliki performa waktu pembacaan selama 32 menit. Rata-rata selisih didapatkan nilai 0,53°C. Nilai tersebut masih wajar diterima karena kemampuan DHT11 yang tidak dapat menghasilkan nilai desimal. Untuk penentuan RH dapat dilihat pada kurva kalibrasi sebagai berikut.

Nilai rata-rata selisih dari perbandingan yang dilakukan didapatkan sebesar 1,87%. Nilai tersebut tidak jauh berbeda dengan pembacaan yang dilakukan tanpa menggunakan *extension*. Untuk persamaan linearitas diperoleh $y = 1,0275x$. Pembacaan RH oleh DHT11 pada rentang 80% - 51% dilakukan selama 32 menit. Performa waktu tersebut saling bergantung antara suhu dan kelembaban relatif sehingga diperoleh waktu yang sama. Kurva perbandingan pada sensor BH1750 menggunakan *extension* disajikan pada **Gambar 5**.



Gambar 5. Kurva perbandingan sensor BH1750 dengan *extension*

Rentang 0 Lux – 12.000 Lux didapatkan performa lamanya pembacaan oleh sensor yakni selama 1 menit 12 detik. Persamaan linearitas dihasilkan $y = 1,0077x$. Untuk rata-rata selisih kedua alat dihasilkan sebesar 64,46%, nilai tersebut dipengaruhi oleh adanya *extension* pada BH1750. Namun, persentase rata-rata nilai *error* hasil pembacaan masih dalam batasan wajar. Persentase *error* berguna dalam penentuan kalibrasi alat *monitoring*. Berikut disajikan tabel rata-rata persentase *error* kalibrasi pengamatan pembacaan kedua sensor dengan perlakuan menggunakan *extension* dan *non-extension* dibandingkan dengan *environment meter*.

Tabel 1. Rata-rata persentase *error* sensor

Sensor	Pengamatan	$\sum Error$ (%)		
		Suhu	RH	Lux
DHT11	<i>Extension</i>	1,74	2,73	-
	<i>Non-extension</i>	1,56	2,95	-
BH1750	<i>Extension</i>	-	-	1,31
	<i>Non-extension</i>	-	-	0,76
Rata-Rata		1,65	2,84	1,04

Terlihat pada **Tabel 1**, hasil pengambilan data DHT11 *non-extension* memiliki nilai rata-rata persentase *error* untuk suhu dan kelembaban relatif sebesar 1,74% dan 2,73%. Sedangkan, hasil data kalibrasi DHT11 dengan menggunakan *extension* diperoleh rata-rata persentase *error* sebesar 1,56% untuk suhu dan 2,95% pada kelembaban relatif. Berdasarkan nilai tersebut diketahui bahwa sensor

DHT11 menggunakan *extension* maupun *non-extension* memiliki nilai kalibrasi yang berdekatan dan nilai persentase *error* yang cukup rendah. Hal tersebut didasarkan pada spesifikasi kemampuan DHT11 keluaran pabrik yang memiliki akurasi sensor suhu kurang lebih 2°C dan sensor kelembaban relatif kurang lebih 5°C. Oleh karena itu, tidak diperlukan penambahan program perintah penambahan nilai kalibrasi pada *software Arduino IDE*. Hasil akhir pengamatan pada percobaan intensitas pencahayaan menggunakan sensor BH1750 diperoleh nilai rata-rata persentase *error* pada sensor *non-extension* sebesar 0,76% dan memiliki *extension* 1,31%. Dengan demikian, diketahui bahwa nilai persentase *error* yang tidak terlalu jauh dan dinilai masih wajar sehingga tidak diperlukan penambahan perintah di *software*.

Penentuan tingkat kenyamanan termal pada prototype ini mengacu SNI-03-6572-2001, dimana tingkat kenyamanan termal dibagi 3 yakni Sejuk Nyaman (SN) jika suhu udara yang terbaca pada rentang 20.5 – 22.8 °C, Nyaman Optimal (NO) jika suhu udara yang terbaca pada rentang 22.8 – 25.8 °C, Hangat Nyaman (HN) jika suhu udara yang terbaca pada rentang 25.8 – 27.1 °C. Jika suhu udara yang terukur di luar tersebut di atas maka prototype akan menampilkan sebagai Tidak Nyaman (TN). Jika pengukuran RH menghasilkan nilai 40 – 50% maka akan ditampilkan sebagai kondisi Nyaman (NM), dan di luar rentang nilai tersebut, hasil pembacaan menampilkan Tidak Nyaman (TN). Pada alat ukur standar yakni *Environment meter*, tingkat kenyamanan tidak dapat ditentukan karena hanya berfungsi sebagai alat pengukur suhu dan RH serta intensitas cahaya. Untuk menentukan tingkat kenyamanan masih tergantung kepada pengetahuan operator alat terhadap baku mutu.

Penentuan tingkat kenyamanan termal juga dapat dilakukan dengan menggunakan metode *Predicted Mean Vote* (PMV) dengan menggunakan aplikasi *CBE Thermal Comfort Tool* yang dikeluarkan oleh ASHRAE-55 (*American Society of Heating, Refrigerating, and Air-Conditioning Engineers*). Namun untuk menggunakan metode ini, data yang diperlukan tidak hanya suhu dan kelembaban relatif udara yang dihasilkan alat ukur biasa, namun memerlukan data-data tambahan sebagai parameter input. Parameter input tersebut adalah kecepatan angin di dalam ruangan, jenis aktivitas yang dilakukan dalam ruangan yg diukur untuk menentukan laju metabolisme tubuh dan jenis pakaian yang digunakan penghuni untuk menentukan nilai koefisien pakaian.pada menggunakan ASHRAE-55 juga diterapkan pada alat *monitoring*. Dalam penentuan kenyamanan termal pada ASHRAE-55 digunakan skala *Predicted Mean Vote* (PMV) dan *Predicted Percentage of Dissatisfied* (PPD). Penentuan tersebut dilakukan menggunakan aplikasi *CBE Thermal Comfort Tool for ASHRAE-55* (*American Society of Heating, Refrigerating, and Air-Conditioning Engineers*). Pengukuran dilakukan di kamar kos dan lobi kos dengan perlakuan menggunakan *extension* dan *non-extension*. Berdasarkan pengamatan yang dilakukan menggunakan prinsip ASHRAE-55 dinilai kurang tepat untuk diaplikasikan pada alat *monitoring* dibandingkan penentuan kenyamanan termal dengan SNI-03-6572-2001 pada *prototype* yang dibuat. Dikarenakan perlunya parameter lain seperti kecepatan angin, kelembaban relatif, nilai insulasi pakaian dan nilai metabolisme sehingga membutuhkan komponen atau sensor lain yang dapat membaca parameter tersebut.

3.4 Penerapan aplikasi Blynk IoT

Penggunaan aplikasi *Blynk IoT* pada penelitian ini berfungsi sebagai *monitoring* jarak jauh terkait hasil pembacaan sensor pada *prototype hardware*. Dengan menggunakan aplikasi *Blynk IoT* yang terpasang pada *smartphone* memungkinkan dapat dilakukan *monitoring* data secara *real time* dalam

bentuk angka maupun grafik [15]. Aplikasi ini terhubung dengan internet melalui *Wi-Fi* dari mikrokontroler ESP01. Karena dalam penggunaannya membutuhkan internet untuk menampilkan hasil maka dapat dikatakan bahwa penelitian ini menerapkan konsep *Internet of Things* (IoT). Hasil yang ditampilkan pada *Blynk* IoT dapat menentukan kenyamanan ruangan secara jarak jauh dengan jangkauan lebih kurang 20 m. Untuk menghubungkan aplikasi *Blynk* IoT dengan hasil pembacaan pada *prototype hardware* dalam penelitian ini menggunakan mikrokontroler ESP01. ESP01 hanya dapat bekerja pada tegangan tidak lebih dari 3,3V apabila disambungkan pada tegangan 5V di pin arduino uno maka terjadi *overheat* yang menyebabkan ESP01 rusak. Namun, penelitian ini menggunakan modul adaptor ESP01 yang dapat terhubung pada tegangan 5V pada *hardware prototype*.

Konsep hasil tampilan pada aplikasi *Blynk* IoT yaitu pembacaan pada serial monitor yang ditangkap dan dikumpulkan lalu di tampilkan pada *widget* di aplikasi *Blynk* IoT. Penerapan aplikasi *Blynk* IoT pada penelitian ini berhasil dilakukan dan dapat ditandai dengan tampilan hasil pada aplikasi *Blynk* IoT dan *website Blynk Cloud*. Tampilan *widget* dapat menampilkan hasil nilai yang sama pada tampilan di LCD.

4. Kesimpulan

Berdasarkan penelitian dan analisis yang telah dilakukan, sistem monitoring dirancang pada software Proteus 8 Professional dan diimplementasikan dalam bentuk hardware prototype. Untuk beroperasi mikrokontroler arduino uno membutuhkan programming melalui software Arduino IDE. Berdasarkan penerapan tersebut, dihasilkan alat sistem monitoring suhu dan kelembaban relatif menggunakan sensor DHT11 dan sensor BH1750 untuk intensitas pencahayaan. Hasil pembacaan sensor terlihat pada tampilan LCD dan *Blynk* IoT. Penentuan tingkat kenyamanan termal berdasarkan pada SNI-03-6572-2001 dan kenyamanan visual bersumber dari SNI-03-6575-2001. Pembacaan tingkat kenyamanan ruangan diatur melalui *software Arduino IDE* menggunakan logika *if else* dan *upload* pada *board* arduino uno. Logika tersebut akan menampilkan tingkat kenyamanan berdasarkan hasil pembacaan sensor sesuai ketentuan SNI. Sistem *monitoring* berbasis mikrokontroler arduino uno memiliki keunggulan dapat menentukan tingkat kenyamanan ruangan yang tidak dimiliki *environment meter*. Hasil rata-rata persentase *error* pembacaan *environment meter* dengan sensor *non-extension* pada suhu, kelembaban relatif, dan intensitas pencahayaan berturut-turut sebesar 1,56%; 2,95%; dan 0,76%. Sedangkan menggunakan *extension* untuk parameter yang sama sebesar 1,74%; 2,73%; 1,31%.

Daftar Pustaka

- [1] Handoko P, Hermawan H, Nasucha M. Sistem kendali alat elektronika menggunakan mikrokontroler Arduino Uno R3 dan *ethernet shield* dengan antarmuka berbasis android. *Jurnal Dinamika Rekayasa*. 2018. 14(2): 92-103.
- [2] Hardianto R, Kusuma C. Rancang bangun *smart lamp* menggunakan micro controller Arduino Uno. *Jurnal Sistem Informasi*. 2019. 1(1):28-37.
- [3] Ganji Kheybari A, Steiner T, Liu S, Hoffmann S. Controlling switchable electrochromic glazing for energy savings, visual comfort and thermal comfort: A model predictive control. *Journal Civil Engineering*. 2021. 2(4):1019-1053.
- [4] Rahmadiansyah A, Orlanda E, Wijaya M, Nugroho HW, Rifki Firmansyah R. Perancangan sistem telemetri untuk mengukur intensitas cahaya berbasis *sensor light dependent resistor* dan Arduino Uno. *Journal of Electrical and Electronic Engineering*. 2017. 1(1):15-21.
- [5] Fauzan MV, Husodo B. Analisis peluang penghematan konsumsi energi pada peralatan listrik di gedung Kantor PT PLN (Persero) UPT Bogor. *Jurnal Teknologi Elektro*. 2020. 11(1):16-23.
- [6] Sumarjono A. Sistem monitoring dan pengendalian suhu ruangan di laboratorium dengan

- menggunakan *Labview* berbasis Arduino. *Integrated Lab Journal*. 2018. 6(2):19-28.
- [7] Budisanjaya IPG, Sucipta IN. Rancang bangun pengendali suhu, kelembaban udara dan cahaya dalam *greenhouse* berbasis Arduino dan android. *Jurnal Ilmiah Teknologi Pertanian Agrotechno*. 2018. 3(2):325-327.
- [8] Suhendri H, Goeritno A. Pemantauan energi listrik pada satu kwh-meter fase tunggal untuk empat kelompok beban berbasis metode payload data handling. *Jurnal Rekayasa Elektrika*. 2018. 14(3): 189-197.
- [9] Sintia W, Hamdani D, Risdianto E. Rancang bangun sistem monitoring kelembaban tanah dan suhu udara. *Jurnal Kumparan Fisika*. 2018. 1(2):60-5.
- [10] [BSN] Badan Standarisasi Nasional. 2001a. Tata cara perancangan sistem ventilasi dan pengkondisian udara pada bangunan gedung, SNI 03-6572-2001. Jakarta: Badan Standarisasi Nasional.
- [11] [BSN] Badan Standarisasi Nasional. 2001b. Tata cara perancangan sistem intensitas cahaya buatan pada bangunan gedung, SNI 03-6575-2001. Jakarta: Badan Standarisasi Nasional.
- [12] Gunawan, Fatimah T. Implementasi sistem pengaturan suhu ruang server menggunakan sensor DHT11 dan sensor PIR berbasis mikrokontroler. *Jurnal Edumatic*. 2020. 4(1):101-110.
- [13] Yuliansya H. Uji kinerja pengiriman data secara wireless menggunakan modul ESP8266 berbasis *rest architecture*. *Jurnal Rekayasa dan Teknologi Elektro*. 2016. 10(2):66-77.
- [15] Indianto W, Kridalaksana AH, Yulianto. Perancangan sistem prototipe pendeteksi banjir peringatan dini menggunakan Arduino dan PHP. *Jurnal Informatika Mulawarman*. 2017. 12(1):45-49.
- [16] Setiawan ET. Pengendalian lampu rumah berbasis mikrokontroler Arduino menggunakan smartphone android. *Jurnal TI-Atma STMIK Atma Luhur Pangkalpinang*. 2015. 1-8.
- [17] Imamsyah, Syaifurrahman, Heindro A. Penerapan aplikasi blynk pada simulator photovoltaic. [Jurnal Skripsi]. Pontianak: Universitas Tanjungpura. 2021.